NanoAgents: Molecular Docking Using Multi-Agent Technology

M.Harindra R. Fernando bquharindra@gmail.com
Department of Computational Mathematics
University of Moratuwa
Sri Lanka

Asoka S. Karunananda askarunananda@gmail.com
Department of Computational Mathematics
University of Moratuwa
Sri Lanka

Roshan P. Perera rpperera@gmail.com
The Institute for Combinatorial Advanced Research and Education
General Sir John Kotelawala Defense University
Sri Lanka

Abstract

Traditional computer-based simulators for manual molecular docking for rational drug discovery have been very time consuming. In this research, a multi agent-based solution, named as NanoAgent, has been developed to automate the drug discovery process with little human intervention. In this solution, ligands and proteins are implemented as agents who pose the knowledge of permitted connections with other agents to form new molecules. The system also includes several other agents for surface determination, cavity finding and energy calculation. These agents autonomously activate and communicate with each other to come up with a most probable structure over the ligands and proteins, which are participating in deliberation. Domain ontology is maintained to store the common knowledge of molecular bindings, whereas specific rules pertaining to the behaviour of ligands and proteins are stored in their personal ontologies. Existing, Protein Data Bank (PDB) has also been used to calculate the space required by ligand to bond with the receptor. The drug discovery process of NanoAgent has exemplified exciting features of multi agent technology, including communication, coordination, negotiation, butterfly effect, self-organizing and emergent behaviour. Since agents consume fewer computing resources, NanoAgent has recorded optimal performance during the drug discovery process. NanoAgent has been tested for the discovery of the known drugs for the known protein targets. It has 80% accuracy by considering the prediction of the correct actual existence of the docked molecules using energy calculations. By comparing the time taken for the manual docking process with the time taken for the molecular docking by NanoAgent, there has been 95% efficiency.

Keywords: Molecular Docking, Multi-agent, Drug Discovery.

1. INTRODUCTION

Molecular docking is a well-recognized branch of research in drug discovery under the field of structural molecular biology. This area is primarily concerned with binding a smaller molecules (ligand) with a target macromolecule (protein). Over many decades, the molecular binding process has been manually modeled by scientist with reference to huge databases of protein and ligand [1]. Nevertheless, the manual molecular docking has been a very time consuming and tedious task. In 1990s, some computer-based solutions for molecular docking have begun to emerge [2]. However, these solutions have appeared as semi-automated approaches to molecular docking, and expected the scientists to use many additional software tools to complete the docking process. As such, automated and integrated software solutions for molecular docking have become a research challenge. This paper present our Multi-Agent Systems based software solution, NanoAgent, for integrated and automated molecular docking. The NanoAgent provides automated all-in-one molecular docking system with the facility to energy minimization, active site prediction, cavity discovery on the target, and orientation calculations. According to experimental results, NanoAgent shows 80% accuracy and 95% efficiency.

The rest of the research paper is organized as follows: Section II critically review the domain of molecular docking by highlighting current solutions, practices, technologies, limitations for defining the research problem. Section III and Section IV describes the essentials of Multi-Agent Technology showing its relevance to solve the molecular docking domain with the aid of Ontology. From the Section V to Section VII, this paper presents our novel approach to molecular docking with Multi-Agent Technology. Section VIII reports on evaluation of the novel solution by explaining evaluation strategy. Section IX concludes the outcome of the research with a note on further work.

2. STATE OF THE ART OF THE MOLECULAR DOCKING

Molecular Docking is usually accomplished among a small molecule and a target macromolecule. There are mainly two types of molecular docking, called “protein-ligand docking” and “protein-protein docking”. Ligand-protein docking is the predominant among the two types of molecular docking [3]. The protein is referred to as the target and also called as the macromolecule. Smaller molecule is called as the drug (or "ligand") is being docked. The goal of molecular docking is to calculate the correct binding mode of a ligand with the protein [4]. Search is performed through the high dimensional spaces efficiently, to find out the correct binding pose. The scoring functions can be used to rank the candidate docking results [5, p. -8] . It is required to evaluate the actual existence of the docked structures using energy values. Most of the time scientists used to do all the above-mentioned calculations manually, for the molecular docking. They have recognized the advantage of the computer to simulate some steps of the manual molecular docking such as energy calculations, cavity finding and surface matching.

The field of molecular docking has appeared during the last decades, driven by the needs of structural molecular biology and structure-based drug discovery [6] using the computer. In computer aided drug design and structural molecular biology heavily uses the molecular docking software tools. In manual molecular docking, scientist needs to get the assistance from the various combinations of separate set of computer-aided tools to generate molecular data including geometries, energies and properties. Manual molecular docking is tedious and time-consuming process due to these reasons.

DOCK 1.0 was the first automated receptor-ligand docking program. It was purposed in 1982 by Irwin Kuntz in The Department of Pharmacology at The University of California at San Francisco [7]. It is used to achieve docking which is the process that two molecules fit together in 3D space. It has the limitations in terms of docking accuracy and time consumption [8]. Molecular docking program, AutoDock [9] was initially written in FORTRAN-77 in 1990 by David S. Goodsell [10] in Arthur J. Olson's laboratory with the assistance of the concepts from DOCK 1.0. It was proposed to achieve automated docking of ligands, which are the small molecules like a candidate drug to their macromolecular targets such as proteins. AutoDock, which is the most famous commercial molecular docking tool [11] in the current computer aided docking context. But it doesn’t have efficient search methods and accurate empirical free energy scoring functions. It also need the assistance from other external tools such as MGLTools [12] and FastGrid [13] to perform some calculations.

The college of Pharmacy in Rosland Franklin University of Medicine and Science in UK [14] which is very famous for the advance chemistry related researches, is relied heavily on computational molecular docking tools. Nevertheless, according to them it is not much easy to use the current available molecular docking tools [15], because they have to access multiple numbers of tools to perform various steps of the docking operations, to the experiment only one single molecular docking process. Scientists from Rosland Franklin University, also mentioned that the regularly, 3D structure data are available for the shape of a protein and a drug independently, but not for the two together [16]. So they have to use-docking tools to visualize the docked molecular structures but they are not efficient and accurate enough [17].

Pattern recognition, machine learning and other artificial intelligence approaches also play a significant role for the screening and identification of candidate molecules [18]. Artificial Neural Network, Fuzzy logic and Genetic Algorithms has been used by most of the drug design in-silico, but it takes considerable amount of time to find the correct orientations of the molecules. And also they need more powerful computer processors to the calculations. In the present most of the molecular docking systems are rather semi-automated, and do not provide integrated solutions. This paper presents our fully automated and integrated molecular docking solution, which exploits the power of Multi Agent Systems and Ontological engineering in modern computing.

3. MULTI-AGENT TECHNOLOGY AND ONTOLOGICAL ENGINEERING

Multi-Agent System technology provides a new paradigm for development of software solutions [19]. This technology supports for the development of software that operates in environments that are distributed and interconnected. Agent technology uses to automating the software tools and the agent behavior relies on the symbolic manipulation [20] of knowledge in order to perform meaningful operations that simulate intelligent abilities. Ontologies can denote domain vocabulary of the agents, the conceptualization that the terms in the vocabulary are intended to capture [21]. For increasing the explanation of the knowledge structure, reducing the language ambiguities and enabling knowledge sharing in software projects are the main usages of the ontologies [22].

3.1 Agent Technology

In the 1990s agent based systems are one of the most inspiring areas of research and development to have emerged in information communication technology [23]. Many researches believe that agents represent the most important new paradigm for software development subsequently object orientation [24]. An agent is a computer system that has the ability of flexible autonomous action in dynamic and unpredictable situations. The performance of Agent is measure based on the environment sequences. Environments are categorized along several dimensions such as Observable, Deterministic, Episodic, Static and Discrete [25]. An agent perceives its environment through sensors and acts on this environment using actuators. Following FIGURE 1 shows the relationship between the Agent and the Environment.

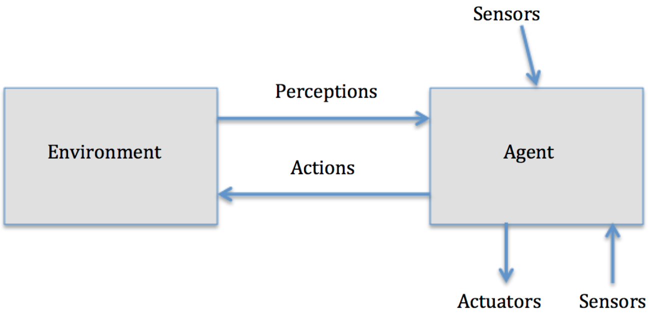

FIGURE 1: PEAS of Agent.

3.2 Characteristics of Software Agent

An agent system is essentially a component system exhibiting several of the characteristics as described follows. There are five characteristics work together to make agent oriented systems more flexible and robust to change [26].

- Adaptability - An agent's behavior may be improved after it has been deployed
- Autonomy - An agent is responsible for its own thread of control and can pursue its own goal largely independent of messages sent from other agents

- Collaboration - Agents communicates and works supportively with other agents to form multi - agent systems working together on some task
- Knowledgeable - An agent is capable of reasoning about its goals and knowledge
- Mobility - The ability for an agent to move from one executing context to another

3.3 Emergent Behavior of Multi-Agent Systems

Emergent behavior is a common phenomenon in complex systems. In both natural and artificial complex systems, individual components perform their actions and make decisions based on local information. But the whole system demonstrates properties and behaviors that have strong global features [27]. Emergent behavior is an fundamental feature in multi-agent systems and plays a important role in the applications of agent technology.

3.4 Distributed Decision Making of MAS

Multi-agent systems (MAS) describe the relationship between several decision-making agents. They are implemented to solve difficult problems in aggregating the strength of each single agent [28]. Hence, multi-agent systems are naturally of a distributed decision making nature. More precisely, MAS is also called distributed artificial intelligence [29] emphasizing the fact that complex problems are solved by a population of different agents. Each of them having its own skills, knowledge, and tasks.

3.5 The Role Of Ontologies

Ontology is a computational model of some portion of the domain [30]. It is a form of a semantic network whose nodes are concepts or objects and its arcs represent relationships or associations among the concepts [31]. It consists of properties and attributes with constraints, functions, and rules that control the behavior of the concepts. Following FIGURE 2 shows the simple ontology that represents relationships and the properties of entities.

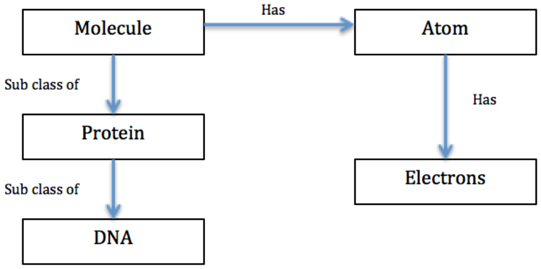

FIGURE 2: Structure of Ontology.

Agent systems are plugged with the domain ontologies. It can be used as the operating rule set [32] of the agents. It is important to exchange and sharing knowledge in multi-agent systems [33] and ontologies are the best way to achieve it.

4. THEORIES BEHIND THE MOLECULAR DOCKING

In molecular docking the small molecule called Ligand usually interconnects with protein's binding sites and also required to follow some rules. Binding sites are areas of protein consider being active in forming of compounds. These are commonly called binding modes. It also forecasts the stability of the binding and the energy of the complex.

4.1 Steps of Molecular Docking

I. Current molecular docking tools follow the common steps as bellow:[34][35]
II. Get the complex coordinates (i.e. from the PDB).
III. Clean the complex (delete all the water and the solvent molecules and all non-interacting ions).
IV. Add the missing hydrogen/side chain atoms and minimized the complex.
V. Clean the minimized complex (delete all the water and the solvent molecules and all non-

interacting ions).

VI. Detached the minimized complex in protein (lock) and ligand (key).
VII. Prepare the docking appropriate files for protein and ligand (pdb files).
VIII. Evaluate the docking results

4.2 Lock and Key Theory

Around 1890, Emil Fischer suggested a model called the lock-and-key model. It explained how biological systems function [36] when docking small molecule with the large molecule. A ligand fits into the active site of a protein, similar as the key fits into a lock is depicted as the following FIGURE 3.

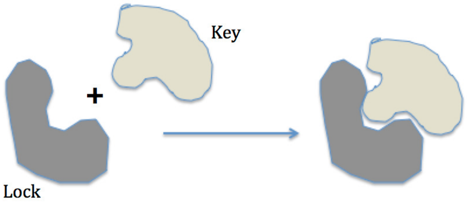

FIGURE 3: Lock and Key Model.

4.3 Energy Minimization

Most of the time the energy minimization analyses are done for the ligand. Ligand can exist in various conformations, which means the shapes or the structures. The conformation, which has minimized energy, will be the most stable conformation of the ligand [37].
There are two types of energies:

Equation (1) is used to calculate Electrostatic Potential Energy [38] . It is Electrostatic Potential Energy a pair wise total of columbic interactions as described in equilibrium.

$$E_{electrostatic} = \sum_{pairs-nonbonded} \frac{q_i q_k}{D_{r_{ik}}} (1)$$

q_i and q_k - the one point charges
r_{ik} – distance between the point charges
D - Coulomb's constant

Equation (2) is to calculate Vander Waals Potential Energy. For general behavior of non-bonded interactions is regularly modeled by the following equation.

$$E_{VanderWaals} = \sum_{pairs-nonbonded} \left(\frac{A_{ik}}{r_{ik}^{12}} - \frac{C_{ik}}{r_{ik}^{6}} \right) (2)$$

r - The distance between two atoms having charges q_i and q_k

Vander Waals potential which expresses the interaction energy using the atom-type dependent constants A and C. Values of A and C may be determined by a variety of methods, like non-bonding distances in crystals and gas-phase scattering measurements.

4.4 Bond Energy

Equation (3) is to calculate the Bond Energy. The energy E, is a function of the atomic locations R, of all the atoms in the structure. These are usually expressed in term of coordinates.

E$_{\text{bonded}}$ - sum of internal, or bonded
E$_{\text{non-bonded}}$ - sum of external or non bonded

$$V(R) = E_{bonded} + E_{nonbonded} \quad (3)$$

Equation (4) is to find the bonded energy. The E$_{\text{bonded}}$ term is a sum of three terms as follows:

$$E_{bonded} = E_{strech} + E_{bend} + E_{rotate} \quad (4)$$

There are three types of atom movements as shown in the following FIGURE 4.

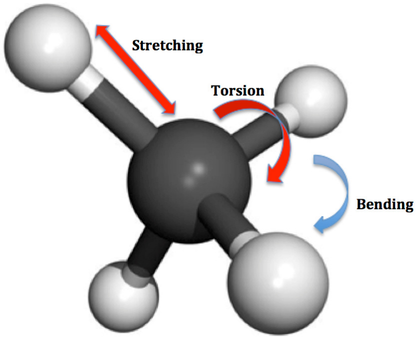

FIGURE 4: Molecular Interactions in Methane Molecule.

4.4.1 Stretching

Equation (5) is to find the bond energy for stretching. This is the approximation of bond energy as a function of b0 and Kb, determines the strength of the bond.

b0 - The ideal bond length
Kb - The force constant

$$E_{streching} = \sum_{1,2\,pairs} K_b \left(b - b_0\right)^2 \quad (5)$$

4.4.2 Bending

Bond energy for bending can be calculated using Equation (6).

θ_0 - harmonic potential
K_θ - Based on chemical type of atoms constituting the angle

$$E_{streching} = \sum_{angles} K_\theta \left(\theta - \theta_0\right)^2 \quad (6)$$

4.4.3 Rotating

Equation (7) can be used to calculate bond energy for rotating.

ϕ - Dihedral angle between each atom axis
Coefficient of symmetry n=1,2,3

$$E_{rotation} = \sum_{1,4\,pairs} K_\phi \left(1 - \cos\left(n\phi\right)\right) \quad (7)$$

4.5 Lipinski's Rule of Five

Lipinski's rule of five [39] is a rule to evaluate drug likeness or determine if a chemical compound with a certain biological activity has properties that would make it a likely orally active drug in living beings. Christopher A. Lipinski articulated it in 1997, based on the inspections that most orally administered drugs are relatively small molecules.

An orally active drug has no more than one disruption of the following:

I. Should not exceed 5 hydrogen bond donors
  a. *The total number of nitrogen–hydrogen and oxygen–hydrogen bonds*
II. Should not exceed 10 hydrogen bond acceptors
  a. *All nitrogen or oxygen atoms*
III. A molecular mass should less than 500 Daltons

According to this rule had chosen the appropriate ligands for the docking.

4.5 Force Field Function

The basic functional form of potential energy in molecular mechanics includes bonded terms for interactions of atoms that are linked by covalent bonds, and non-bonded, terms that describe the long-range electrostatic and van der Waals forces. Following equation (8) [40] is used to calculate the total of the above-mentioned energies:

$$E = \sum_{i} \sum_{j} \left(\frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^{6}} + \frac{q_i q_j}{\varepsilon(r_{ij}) r_{ij}} \right) \qquad (8)$$

r_{ij} - The distance between protein atom i and ligand atom j
A_{ij} and B_{ij} - The VDW parameters
q_i and q_j - The atomic charges
$\varepsilon(r_{ij})$ is usually set to $4r_{ij}$, reflecting the screening effect of water on electrostatic interactions

5. MAS APPROACH TO MOLECULAR DOCKING

Our new intelligent multi-agent systems technology based molecular docking solution has been named as ‘NanoAgents’. The proposed system get the Protein Data Bank (PDB) files as the inputs of the ligand and the proteins and output the discovered structures of the matched protein-ligand pairs. Ligands and proteins are programmed as the agents of the Multi-Agent System and the rule set for the binding codified as domain ontology. Each agent operated according to the rule set. There are the other agents, who help for the docking of LigandAgents and ProteinAgent such as EnergyCalculationAgent, ActiveSiteFindingAgent, SurfaceMatchingAgent and CavityFindingAgent.

5.1 Inputs

Multi-Agent system for Molecular Docking has been design to accept multiple inputs coming from different entities of the Molecular Docking process. The following are the main inputs to the system. TABLE 1 shows the inputs from corresponding entities.

Input	Entity (Agent)
Rules for binding	Ontology Agent
Description of protein	Protein Agent
Description of ligand	Ligand Agent
Description of bonds	Energy Calculation Agent
3D coordinates of the protein	Active Site Finding Agent
3D coordinates of the ligand	Surface Matching Agent
3D coordinates of the ligand and protein	Cavity Finding Agent

TABLE 1: Inputs for the Molecular Docking System.

5.2 Outputs

The output of the system will be the docked molecule. These outputs are coming as the rational drug with its PDB file structure.

5.3 Process for the Molecular Docking

System will use the molecular of database and the ontology as the inputs to generate the rational drugs. In this processing, two major types of agents namely protein agents and ligand agents are defined into the system. The knowledge requires to these agents to operate are stored in common domain ontology and personal PDB files. Get the complex coordinates (i.e. from the PDB) as the input to the system and clean the complex (delete all the water and the solvent molecules and all non-interacting ions).

As the next step, add the missing hydrogen/side chain atoms and minimized the complex. Then clean the minimized complex (delete all the water and the solvent molecules and all non-interacting ions). Separate the minimized complex in macromolecule (lock) and ligand (key) should be done. Then prepare the docking suitable files for LOCK and KEY (pdb files). Then perform the docking and analyze the docking results. There are separate ontologies to store the atom colors and charges for the atoms/elements. Agents get the help from these ontologies to find the best ligand agent to bind with the protein agent.

5.4 JADE – Java Agent Development Framework

JADE (Java Agent DEvelopment Framework) [41] is a software Framework completely implemented in the Java language. It make easy the implementation of multi-agent systems through a middleware that compatible with the FIPA [42] specifications. JADE is completely implemented in Java language and the minimal system requirement is the version 5 of JAVA (the run time environment or the JDK).

5.5 Jena – A Semantic Web Framework for Java

Jena [43] is a Java framework for building Semantic Web applications. It provides a development environment for RDF [44], RDFS [45] and OWL [46], SPARQL [47] and includes a rule - based inference engine.

5.6 Jmol: An Open-Source Java Viewer for Chemical Structures

Jmol [48] is an open-source viewer for three-dimensional chemical structures. It is used as a research tool in chemistry and bio-chemistry. It is free and open source software written in Java programming language and so it runs on any operating system.

6. ANALYSIS AND DESIGN OF THE NANOAGENTS

NanoAgents system design comprises of different system components for different tasks such as, find the active site of the receptor, find the shape of the cavity of the receptor, search matching ligand to the cavity, find correct orientation of the cavity, calculate binding energies and energy minimization and evaluate the rationality of the selected ligand or the drug.

6.1 System Integration of the NanoAgents

The main architecture behind this solution is the blackboard architectural model that has Request Agents, Resource Agents, Message Space and Domain Ontology. AI researchers conceived the concept of blackboard architecture [49] in the 1970's.A message space, which is the blackboard, has all the solutions given by the agents. The common knowledge base, the "blackboard", is iteratively updated by a diverse group of specialist knowledge sources. The overall system architecture and the control flow are showed as following Fig 4.

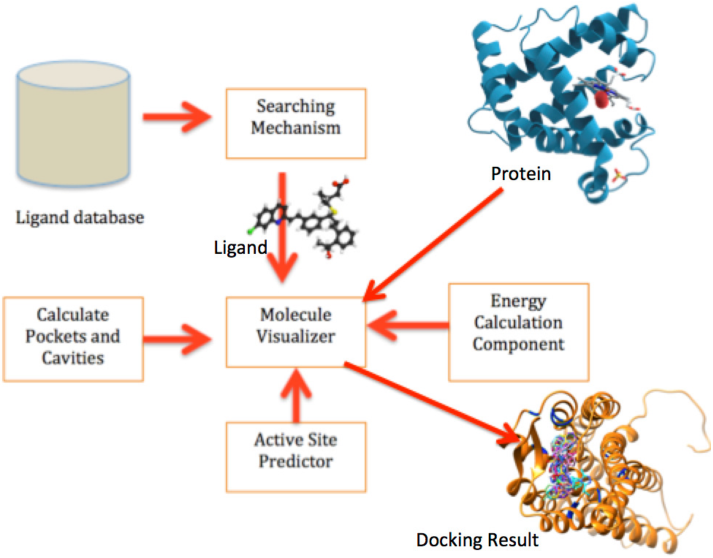

FIGURE 5: System Architecture.

Usually it starts with a problem description and ending with a solution. Each knowledge source revises the blackboard with a partial solution when its internal restrictions match the blackboard state and the specialists work together to solve the problem. The blackboard model was originally designed as a way to handle complex, ill-defined problems, where the solution is the sum of its parts. Molecular docking can be consider as a complex problem which the solution is the sum of different operations, such as energy calculations, active site finding, surface matching and cavity finding.

7. DETAILED DESIGN AND IMPLEMENTATION

In this section let us discuss the software realization of our designed system. And also in here the system flows and algorithm level realizations will be discussed. Implementation comprises of the multi-agent systems based molecular docking system.

7.1 DOCK Algorithm To Find Matching Ligand To The Protein

DOCK is another molecular docking program currently using in the drug discovery field. Following are the steps it follows to find the matching ligand, achieve the docking [50]. And also we have followed their approach similarly.

I. A set of spheres is created inside the active sties of the protein
II. The spheres represent the volume which could be occupied by the ligand
III. Ligand is represented by spheres inside the ligand (see FIGURE 6)

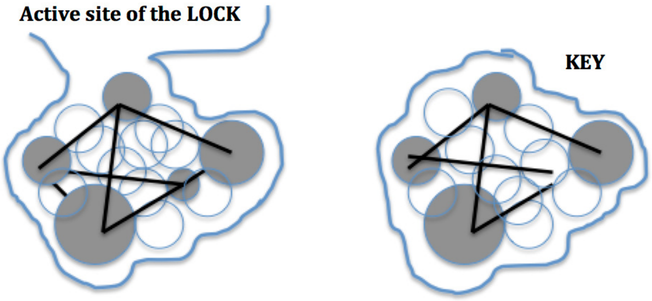

FIGURE 6: DOCK program algorithm: Match the spheres shown in gray (distances between them are used for scoring).

7.2 Methods to Find Active Sites of The Protein

7.2.1 Alpha Spheres

Alpha shape [51] of a set of weighted vertices. It is a concept from computational geometry that has found its use in computational chemistry. The sphere that contacts some atoms on its boundary and contains no internal atoms is called a contact sphere. An alpha sphere is a distinctive case of a contact sphere. An alpha sphere is a sphere that contacts four atoms on its boundary and contains no internal atoms as FIGURE 7.

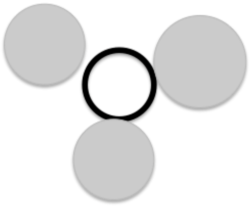

FIGURE 7: An Alpha sphere in 2D that contacts three atoms of different radius.

7.2.2 Finding Cavities in The Protein

One of many properties we can study is a molecular cavity, where cavity is understood as a free space inside a molecule. After reviewing the literature, we are usually interested only in a certain subset of cavities. Voronoi diagram is then used to achieve the task of finding cavities for varying radiuses [52]. In this research it is used this method to find cavities in the target molecule.

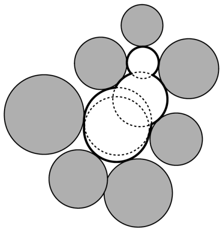

FIGURE 8: Cavities Formed by Gray Atoms.

8. EVALUATION

8.1. Evaluation Methods

For the evaluation of the tool, compared known ligand-protein pair docking time taken in the usual manner with this software tool. Also analyzed the time difference for measuring the efficiency of the tool. The accuracy of the tool had been measured by the energy calculations. It will help to find the actual existence of the ligand-protein pairs in the environment found by the docking tool.

Some of the following ligand and pairs found from the reference [53] had been used to the evaluation of the docking tool as the sample population.

Protein	Ligand
1A52	OHT
1A52	EST
2AM9	B5R
1A52	GEN
1EWV	KAI
4PVU	BRL
4P6W	DEX

TABLE 2: Known Protein-Ligand Pairs.

To compare the results, used that time taken for the manual molecular docking for each protein and ligand pairs. The next step was tracking the time taken for each docking process for the same protein-ligand pairs taken by the developed tool. To evaluate the accuracy of the software tool, also noted that the prediction of the actual existence of each docked pair using the tool's energy calculation module.

8.2 Test Results

Following are the two criteria used to evaluate the tool as explained the above paragraph:

Criteria 01:
The time taken for the molecular docking tool and the manual molecular docking
Criteria 02:
The prediction of the actual existence of the docked molecule

Precision for Criteria 01:
Considered the time taken for the molecular docking tool and the manual molecular docking for the evaluation of the tool.

Probability of success (time taken by the software is lesser than the actual manual molecular docking) = 19/20 = 0.95

Probability of failure (time taken by the software is higher than the actual manual molecular docking) = 1/20 = 0.05

Accuracy of the software = 9/20 = 0.95 = 95%

Precision for Criteria 02:
Considered the prediction accuracy of the actual existence of the docked molecules by the software tool by using energy calculation.

Probability of success (software predicts the actual existence of the docked molecule correctly) = 16/20 = 0.8

Probability of failure (software not predicts the actual existence of the docked molecule correctly) = 4/20 = 0.2

Accuracy of the software = 16/20 = 0.8 = 80%

Following FIGURE 8 shows the main functions of the molecular docking tool and the interfaces. Considering the main features of the software it is an all-in-one solution for the molecular docking comparing other popular tool AutoDock.

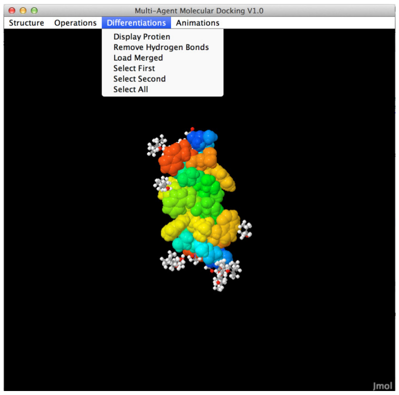

FIGURE 8: Shows the Cavities of The Protein.

Following FIGURE 9 shows the ligand and the receptor on the same window.

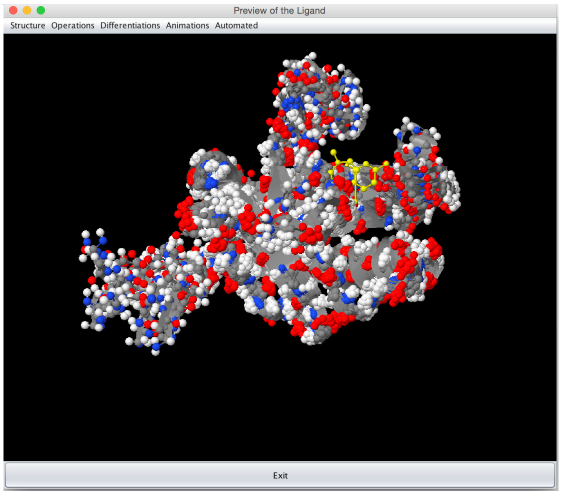

FIGURE 9: Ligand and Receptor.

9. CONCLUSION AND FUTURE WORK

In this research we could show multi-agent technology can be successfully applying to automated the manual molecular docking. The time taken to dock the molecules using our tool was compared with the popular tool called AutoDOCK. And the results were significant.

I have executed the molecular simulation 20 times for the known ligand-protein pairs and compared it with the actual time taken for the manual molecular docking. The experiment results derived based on two scenarios.

Time for the docking: By comparing the time taken for the manual docking process with the time taken for the automated molecular docking, it says the software tool can obtain 95% accuracy
Accuracy: It has 80% accuracy by considering the prediction of the correct actual existence of the docked molecules according to the energy calculations.

So it is evident that the software can manage a good balance in between the efficiency and the accuracy.

As the future work we can improve the tool using Artificial Neural Network by training the system. It will again enhance the efficiency of the tool by reducing the time taken for the docking process. Currently in this research, it was focused on the ligand-protein docking only, but it can be extended to the protein-protein docking as well. This tool can be used by the researches in Biological Chemistry domain. Hope to get their feed back to improve the accuracy of the software further.

10. REFERENCES

[1] D. T.-H. Chang, C.-H. Ke, J.-H. Lin, and J.-H. Chiang, “AutoBind: automatic extraction of protein–ligand-binding affinity data from biological literature,” Bioinformatics, vol. 28, no. 16, pp. 2162–2168, Aug. 2012.

[2] M. Munetomo, K. Akama, and H. Maeda, “An automated ligand evolution system using Bayesian optimization algorithm,” WSEAS Trans. Inf. Sci. Appl., vol. 6, no. 5, pp. 788–797, 2009.

[3] S. S. Azam and S. W. Abbasi, "Molecular docking studies for the identification of novel melatoninergic inhibitors for acetylserotonin-O-methyltransferase using different docking routines," Theor. Biol. Med. Model., vol. 10, p. 63, Oct. 2013.

[4] G. M. Morris and M. Lim-Wilby, "Molecular docking," in Molecular Modeling of Proteins, Springer, 2008, pp. 365–382.

[5] M. A. D. JE, "Virtual screening and lead optimisation to identify novel inhibitors for HDAC-8," ArXiv Prepr. ArXiv12092793, 2012.

[6] V. Lounnas, T. Ritschel, J. Kelder, R. McGuire, R. P. Bywater, and N. Foloppe, "CURRENT PROGRESS IN STRUCTURE-BASED RATIONAL DRUG DESIGN MARKS A NEW MINDSET IN DRUG DISCOVERY," Comput. Struct. Biotechnol. J., vol. 5, no. 6, pp. 1–14, Feb. 2013.

[7] "Molecular Docking to Ensembles of Protein Structures, Ronald M.A. Knegtel,Irwin D.Kuntz,Department of PharmaceuticalChemistry School of Pharmacy University of California, San Francisco, CAand, C.M.Oshiro, 1997." .

[8] Z. Zhou, A. K. Felts, R. A. Friesner, and R. M. Levy, "Comparative Performance of Several Flexible Docking Programs and Scoring Functions: Enrichment Studies for a Diverse Set of Pharmaceutically Relevant Targets," J. Chem. Inf. Model., vol. 47, no. 4, pp. 1599–1608, 2007.

[9] "AutoDock — AutoDock." [Online]. Available: http://autodock.scripps.edu/. [Accessed: 15-Jul-2014].

[10] "Home of David S. Goodsell." [Online]. Available: http://mgl.scripps.edu/people/goodsell/. [Accessed: 15-Sep-2014].

[11] M. Lape, C. Elam, and S. Paula, "Comparison of current docking tools for the simulation of inhibitor binding by the transmembrane domain of the sarco/endoplasmic reticulum calcium ATPase," Biophys. Chem., vol. 150, no. 1–3, pp. 88–97, Aug. 2010.

[12] "MGLTools Website - Welcome — MGLTools." [Online]. Available: http://mgltools.scripps.edu/. [Accessed: 15-Sep-2014].

[13] M. Olšák, J. Filipovič, and M. Prokop, "FastGrid–The Accelerated AutoGrid Potential Maps Generation for Molecular Docking," Comput. Inform., vol. 29, no. 6+, pp. 1325–1336, 2012.

[14] "Laboratory for Molecular Design and Drug Discovery." [Online]. Available: http://www.rosalindfranklin.edu/cop/Home/LaboratoryforMolecularDesignandDrugDiscovery.aspx. [Accessed: 25-Sep-2014].

[15] C. Ma, R. Kotaria, J. A. Mayor, S. Remani, D. E. Walters, and R. S. Kaplan, "The Yeast Mitochondrial Citrate Transport Protein CHARACTERIZATION OF TRANSMEMBRANE DOMAIN III RESIDUE INVOLVEMENT IN SUBSTRATE TRANSLOCATION," J. Biol. Chem., vol. 280, no. 3, pp. 2331–2340, Jan. 2005.

[16] C. A. Nicolaou and N. Brown, "Multi-objective optimization methods in drug design," Drug Discov. Today Technol., vol. 10, no. 3, pp. e427–e435, Sep. 2013.

[17] P. H. Palestro, L. Gavernet, G. L. Estiu, and L. E. Bruno Blanch, "Docking Applied to the Prediction of the Affinity of Compounds to P-Glycoprotein," BioMed Res. Int., vol. 2014, p. e358425, May 2014.

[18] W. Duch, K. Swaminathan, and J. Meller, "Artificial intelligence approaches for rational drug design and discovery," Curr. Pharm. Des., vol. 13, no. 14, pp. 1497–1508, 2007.

[19] E. Yu, “Agent-oriented modelling: software versus the world,” in Agent-Oriented Software Engineering II, Springer, 2002, pp. 206–225.

[20] G. D. Crnkovic, “The Cybersemiotics and Info-Computationalist Research Programmes as Platforms for Knowledge Production in Organisms and Machines,” Entropy, vol. 12, no. 4, pp. 878–901, Apr. 2010.

[21] B. Chandrasekaran, J. R. Josephson, and V. R. Benjamins, “What are ontologies, and why do we need them?,” IEEE Intell. Syst., vol. 14, no. 1, pp. 20–26, 1999.

[22] A. H. André Filipe de Moraes Batista, Emerson Aguiar Noronha, Fábio Aragão da Silva, Maria das Graças Bruno Marietto, Robson dos Santos França Terry Lima Ruas, Modeling Artificial Life Through Multi-Agent Based Simulation. INTECH Open Access Publisher, 2011.

[23] M. University of Southampton Luck and AgentLink, Agent technology: computing as interaction ; a roadmap for agent based computing. Southampton: University of Southampton on behalf of AgentLink III, 2005.

[24] “‘Agent Technology : Computing as Interaction,A Roadmap for Agent Based Computing’.[Online]. Available: http://www.das.ufsc.br/~jomi/mas/leituras/agentlink-roadmap.pdf [Accessed: 16-Sep-2014].” .

[25] D. J. Cook, “Multi-agent smart environments,” J. Ambient Intell. Smart Environ., vol. 1, no. 1, pp. 51–55, 2009.

[26] Q. Guo and M. Zhang, “Research on intelligent manufacturing system based on multi-agent,” in Intelligent Robotics and Applications, Springer, 2008, pp. 829–838.

[27] Z. Li, C. H. Sim, and M. Y. H. Low, “A survey of emergent behavior and its impacts in agent-based systems,” in Industrial Informatics, 2006 IEEE International Conference on, 2006, pp. 1295–1300.

[28] “Wooldridge M. and Jennings N. ‘Intelligent Agents: Theory and Practice’.1995.” .

[29] G. Weiss, Multiagent systems a modern approach to distributed artificial intelligence. Cambridge, Mass.: MIT Press, 1999.

[30] “An Ontology for Engineering Mathematics.” [Online]. Available: http://www-ksl.stanford.edu/knowledge-sharing/papers/engmath.html. [Accessed: 03-Aug-2015].

[31] M. N. Huhns and M. P. Singh, “Ontologies for agents,” Internet Comput. IEEE, vol. 1, no. 6, pp. 81–83, 1997.

[32] F. Zini and L. Sterling, “(1) DISI-Universit a di Genova via Dodecaneso 35, 16146 Genova, Italy zini@ disi. unige. it (2) Department of Computer Science & Software Engineering.”

[33] P. Maret, J. Calmet, and others, “Agent-based knowledge communities,” Int. J. Comput. Sci. Appl., vol. 6, no. 2, pp. 1–18, 2009.

[34] “Molecular_Docking_Tutorial.pdf.” .

[35] “Machado_et_al_LNCS_4643_1-11_2007.pdf.” .

[36] “Carboxypeptidase.” [Online]. Available: http://www.chemistry.wustl.edu/~edudev/LabTutorials/Carboxypeptidase/carboxypeptidase.html. [Accessed: 17-Sep-2014].

[37] "Potential Energy Function." [Online]. Available: http://www.ch.embnet.org/MD_tutorial/pages/MD.Part2.html. [Accessed: 17-Sep-2014].

[38] "Electric energy and potential." [Online]. Available: http://physics.bu.edu/~duffy/PY106/Potential.html. [Accessed: 25-Oct-2014].

[39] P. Leeson, "Drug discovery: Chemical beauty contest," Nature, vol. 481, no. 7382, pp. 455–456, Jan. 2012.

[40] J. W. Ponder and D. A. Case, "Force fields for protein simulations," Adv. Protein Chem., vol. 66, pp. 27–85, 2003.

[41] "Jade Site | Java Agent DEvelopment Framework." [Online]. Available: http://jade.tilab.com/. [Accessed: 17-Sep-2014].

[42] "FIPA ACL Message Structure Specification." [Online]. Available: http://www.fipa.org/specs/fipa00061/SC00061G.html. [Accessed: 18-Jul-2014].

[43] "Apache Jena - Home." [Online]. Available: https://jena.apache.org/. [Accessed: 09-Sep-2014].

[44] "RDF Current Status - W3C." [Online]. Available: http://www.w3.org/standards/techs/rdf#w3c_all. [Accessed: 29-Jul-2014].

[45] Y. Yan, C. Wang, A. Zhou, W. Qian, L. Ma, and Y. Pan, "Efficiently querying rdf data in triple stores," in Proceedings of the 17th international conference on World Wide Web, 2008, pp. 1053–1054.

[46] J. C. Arpírez, A. Gómez-Pérez, A. Lozano-Tello, and H. S. A. N. Pinto, "Reference ontology and (ONTO) 2 agent: the ontology yellow pages," Knowl. Inf. Syst., vol. 2, no. 4, pp. 387–412, 2000.

[47] "SPARQL Query Language for RDF." [Online]. Available: http://www.w3.org/TR/rdf-sparql-query/. [Accessed: 25-Jul-2014].

[48] "Jmol: an open-source Java viewer for chemical structures in 3D." [Online]. Available: http://jmol.sourceforge.net/. [Accessed: 17-Sep-2014].

[49] D. Rudenko and A. Borisov, "An overview of blackboard architecture application for real tasks," in Scientific Proceedings Of Riga Technical University, Ser, 2007, vol. 5, pp. 50–57.

[50] T. J. Ewing and I. D. Kuntz, "Critical evaluation of search algorithms for automated molecular docking and database screening," J. Comput. Chem., vol. 18, no. 9, pp. 1175–1189, 1997.

[51] "Locating Binding Sites in Protein Structures." [Online]. Available: http://www.chemcomp.com/journal/sitefind.htm. [Accessed: 08-Sep-2014].

[52] L. Jirkovskỳ, S. M. Maňák, and I. Kolingerová, "Finding Cavities in a Molecule".

[53] "Latest pairings | IUPHAR/BPS Guide to PHARMACOLOGY." [Online]. Available: http://www.guidetopharmacology.org/latestPairings.jsp. [Accessed: 17-Sep-2014].